Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1.   (Previously presented) A computer program product, tangibly embodied in a machine-readable storage device comprising instructions operable to cause data processing apparatus to:

implement a reusable software component encapsulating functionality, multiple instances of the component being usable at the same time;

the reusable component having at least one visual representation;

the reusable component having a programming interface for programmatic interaction with the reusable component;

the reusable component having a data-binding interface for data communication with the reusable component; and

the reusable component having a visual interface for access to the at least one visual representation of the reusable component.

2.   (Original) The computer program product of claim 1, wherein the programming interface, the data-binding interface, and the visual interface are separate interfaces.

3.   (Previously presented) The computer program product of claim 1, further comprising instructions to implement one or more controllers for the reusable component, each controller having an associated context for storing data and state for the controller.

4.   (Currently amended) The computer program product of claim 3, further comprising instructions to implement one or more views for the reusable component, each view providing a visual representation of the <u>reusable</u> component.

5. (Previously presented) The computer program product of claim 1, further comprising instructions to:

    embed a sub-component into the reusable component.

6. (Original) The computer program product of claim 5 wherein the instructions to embed a sub-component comprise instructions to:

    use a programming interface, a data-binding interface, and a visual interface of the sub-component.

7. (Original) The computer program product of claim 1 wherein the programming interface includes an interface controller having an interface controller context and a configuration controller having a configuration controller context, the visual interface includes an interface view, and the data-binding interface provides context mapping for the interface controller context and the configuration context.

8. (Previously presented) A computer program product, tangibly embodied in a machine-readable storage device, the computer program product comprising instructions operable to cause data processing apparatus to:

    implement an application runtime framework, the framework being operable to:

        receive a specification of a component interface to be used in an application without a specification of a corresponding component implementation, the component interface having a programming interface, a data-binding interface, and a visual interface; and

        instantiate a particular component implementation at application runtime, the particular component implementation being selected from one or more component implementations corresponding to the component interface.

9. (Cancelled)

10. (Previously presented) A computer program product, tangibly embodied in a machine-readable storage device, for implementing an application runtime framework, the computer

program comprising instructions operable to cause data processing apparatus to:

receive an event subscription directed to a subscribing component when the subscribing component has not been initiated, the event subscription specifying subscriptions to one or more events generated by sub-components embedded by the subscribing component;

cache events generated by the sub-components that are specified by the event subscription while the subscribing component has not been instantiated; and

forward any cached events to an instance of the subscribing component after the subscribing component is instantiated.

11. (Currently amended) A computer program product, tangibly embodied in a machine-readable storage device, for implementing an application runtime framework, the computer program product comprising instructions operable to cause data processing apparatus to:

receive one or more context mappings for a component, the context mappings being specified by a component embedder to exchange context data with the component, the context data comprising interface ~~context~~ data and configuration ~~context~~ data;

if the component has not been instantiated, cache the specified context mappings; and

create the specified context mappings for the component after the component has been instantiated.

12. (Previously presented) A computer implemented method, comprising:

implementing a reusable software component encapsulating functionality, multiple instances of the component being usable at the same time;

the reusable component having at least one visual representation;

the reusable component having a programming interface for programmatic interaction with the reusable component;

the reusable component having a data-binding interface for data communication with the reusable component;

the reusable component having a visual interface for access to the at least one visual

representation of the reusable component; and

storing the reusable component.

13. (Previously presented) The method of claim 12, wherein implementing the reusable component comprises:

implementing the reusable component having the programming interface, the data-binding interface, and the visual interface as separate interfaces.

14. (Previously presented) The method of claim 12, further comprising:

implementing one or more controllers for the reusable component, each controller having an associated context for storing data and state for the controller.

15. (Previously presented) The method of claim 14, further comprising:

implementing one or more views for the reusable component, each view providing a visual representation of the reusable component.

16. (Previously presented) The method of claim 12, further comprising:

embedding a sub-component into the reusable component.

17. (Original) The method of claim 16, wherein embedding the sub-component comprises:

using a programming interface, a data-binding interface, and a visual interface of the sub-component.

18. (Previously presented) The method of claim 12, wherein implementing the reusable component comprises:

implementing the reusable component having the programming interface including an interface controller having an interface controller context and a configuration controller having a configuration controller context, the visual interface including an interface view, and the data-binding interface providing context mapping for the interface controller context and the configuration context.

19. (Previously presented) An apparatus, comprising:

means for implementing a reusable software component encapsulating functionality, multiple instances of the component being usable at the same time;

the reusable component having at least one visual representation;

the reusable component having a programming interface for programmatic interaction with the reusable component;

the reusable component having a data-binding interface for data communication with the reusable component; and

the reusable component having a visual interface for access to the at least one visual representation of the reusable component.